

This Page Is Inserted by IFW Operations  
and is not a part of the Official Record

## **BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning documents *will not* correct images,  
please do not report the images to the  
Image Problem Mailbox.**

Publication date:

1998-06-02

Inventor(s):

KUREYAMA NOBUO; MATSUDA YOSHIKI; HASHIMOTO TETSUYA;  
AOSHIMA TATSUTO

Applicant(s)::

HITACHI LTD

Requested Patent:

JP10149326

Application Number:

JP19960322205 19961118

Priority Number(s):

IPC Classification:

G06F13/00

EC Classification:

Equivalents:

#### Abstract

**PROBLEM TO BE SOLVED:** To provide a remote program-user interaction method making it unnecessary to maintain the connection of a network in interactive processing in user's interaction with a program to be driven by a server computer different from a client computer to be operated by the user and allowed to be used even in the case of using a communication means having low instantaneous response such as an electronic mail.

**SOLUTION:** A process is managed by a software unit 301 constituted of a main processing program 312, a user interface display program 313, a data block to be referred to and updated from respective programs, and management information 311. In the case of interacting with a user, the program 313, the data block 321 referred to and updated from the program 313 and the information 311 are transmitted to a client computer operated by the user and executed and information obtained by user interaction is stored in the data block 321. After the end of the interaction, the software unit 301 is returned to a server and the main processing program 312 executes operation by referring to the updated data block 321.

Data supplied from the esp@ccnet database - 12

# 01F19

HDR 0044046

Copies: **1**

ORDER JP 10-149326 ISSUE DATE 6 02 1998 PAGES ALL  
ISSN -  
YEAR -

/0044046/

**CPL**

JD PALMER  
IBM CORPORATION  
Dept QPZ Zip MD90/201  
Bldg 201-1 Room 1J25A  
8501 IBM DR  
CHARLOTTE, NC 28262-8563



November 10, 2000



YI1950

\*777 T 6330\*

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平10-149326

(43) 公開日 平成10年(1998) 6月2日

(51) IntCl<sup>5</sup>

G 0 6 F 13/00

識別記号

3 5 7

P I

G 0 6 F 13/00

3 5 7 Z

審査請求 未請求 請求項の数4 F D (全 15 頁)

(21) 出願番号

特願平8-322205

(22) 出願日

平成8年(1996)11月18日

(71) 出願人

000005108

株式会社日立製作所

東京都千代田区神田駿河台四丁目6番地

(72) 発明者

紅山 伸夫

神奈川県川崎市幸区鹿島田890番地 株式会社日立製作所情報・通信開発本部内

(72) 発明者

松田 芳樹

神奈川県川崎市幸区鹿島田890番地 株式会社日立製作所情報・通信開発本部内

(72) 発明者

橋本 哲也

神奈川県川崎市幸区鹿島田890番地 株式会社日立製作所情報・通信開発本部内

(74) 代理人

弁理士 矢島 保夫

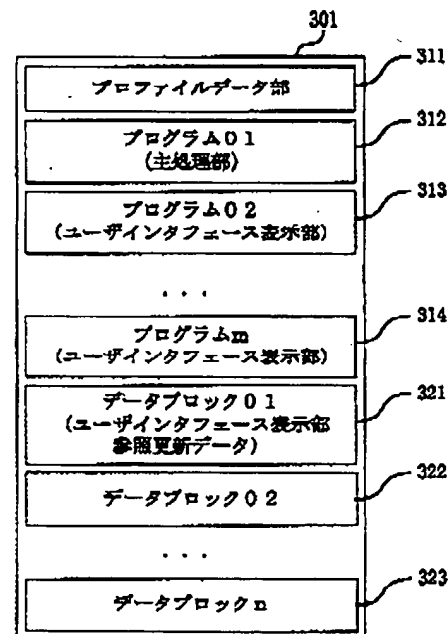
最終頁に続く

(54) 【発明の名称】 リモートプログラムユーザ対話方法

(57) 【要約】

【課題】ユーザが操作するクライアント計算機と異なるサーバ計算機で動作するプログラムとのユーザ対話において、対話処理中にネットワーク接続の維持が不要で、電子メール等の即時応答性の低い通信手段を用いても利用可能なリモートプログラムユーザ対話方法を提供する。

【解決手段】主処理プログラム312とユーザインタフェース表示プログラム313と各プログラムから参照更新可能なデータブロック321及び管理情報311とから構成されるソフトウェア単位301でプロセスを管理する。ユーザとの対話を行う際には、ユーザの操作するクライアント計算機へユーザインタフェース表示プログラムと該プログラムから参照更新されるデータブロックと管理情報とを送信して実行し、ユーザ対話で得た情報を該データブロックへ格納する。対話終了後、該ソフトウェア単位はサーバへ返送され、主処理プログラムは更新されたデータブロックを参照し動作する。



(2)

特開平10-149326

## 【特許請求の範囲】

【請求項1】異なる計算機であるサーバ計算機とクライアント計算機とをネットワークに接続すると共に、各計算機はネットワークを介してプログラムやデータを相互に転送するための送受信手段を備えたシステムに適用するリモートプログラムユーザ対話方法であって、前記サーバ計算機で実行される主処理プログラム及び前記クライアント計算機で実行されるユーザインタフェース表示プログラムからなる複数プログラムと、それらのプログラム実行時に参照変更される複数データブロックと、管理情報を保持するプロファイルデータ部とから構成される、エージェントと呼ぶソフトウェア単位を定義して予め前記サーバ計算機側に用意しておき、ユーザの操作する前記クライアント計算機において、前記サーバ計算機で実行される主処理プログラムとの対話を行う場合に、

前記サーバ計算機から前記クライアント計算機に、エージェント内のユーザインタフェース表示プログラムと、該ユーザインタフェース表示プログラム実行時に参照変更されるデータブロックと、プロファイルデータ部とを備えたサブエージェントを送受信手段により送信し、該サブエージェントを受信した前記クライアント計算機では、該サブエージェントのユーザインタフェース表示プログラムをクライアント内で実行してユーザインタフェース画面を表示することでユーザと直接対話し、ユーザ操作により得られた情報を該サブエージェントのデータブロックへ保存し、その後、該サブエージェントを前記サーバ計算機へ返送し、

前記サーバ計算機では、返送された該サブエージェントのデータブロックに格納された変更後のデータを用いて、主処理プログラムを実行することを特徴とするリモートプログラムユーザ対話方法。

【請求項2】前記エージェントの複数プログラムは、各々が異なる処理系で実行されるが同一の動作を行う異なる形式で記述された複数のユーザインタフェース表示プログラムを含み、

前記サーバ計算機から前記クライアント計算機にサブエージェントを送信する際には、前記クライアント計算機で実行可能な形式のユーザインタフェース表示プログラムをサブエージェントに含めて転送することを特徴とする請求項1に記載のリモートプログラムユーザ対話方法。

【請求項3】前記エージェントのデータブロックは、当該エージェントの動作属性定義情報を含み、主処理プログラムの動作の前記クライアント計算機での設定変更を行うことによりプログラム動作カスタマイズを行うことを特徴とする請求項1または2に記載のリモートプログラムユーザ対話方法。

【請求項4】請求項1から3のいずれか1つに記載のリモートプログラムユーザ対話方法を実現するプログラム

を記憶したことを特徴とする記憶媒体。

## 【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、リモートプログラムユーザ対話方法に関し、詳しくは、ユーザが直接操作する計算機と異なる計算機上で動作するプログラムの、動作属性の設定編集やプログラムの保持するデータのユーザへの表示等、ネットワークを介したプログラムとユーザとの対話の実行方法に関する。

【0002】

【従来の技術】従来、複数の計算機がネットワークで接続されたシステムにおいて、ユーザの操作するクライアント計算機において他の計算機で動作するプログラムとのユーザ対話を行う方法としては、以下のものが挙げられる。

【0003】X-Window（「入門X-Window」、アスキー出版等参照）等のシステムでは、サーバクライアント間で通信セッションを確立した後、サーバで動作するプログラムがクライアント計算機へユーザインタフェース画面表示情報を送信し、クライアントからサーバへはユーザが画面に対して行った操作のイベントを送信し、イベントとそれによるフィードバックで更新されるユーザインタフェース画面表示情報とをサーバクライアント間でやり取りすることで対話を進める（以下、従来方法1と呼ぶ）。

【0004】別の方法として、クライアントで動作するユーザインタフェース表示プログラムをネットワークを介してサーバから一時的にダウンロードし、サーバで動作するプログラムとクライアント内で実行されるプログラムとの間で通信を行う（以下、従来方法2と呼ぶ）。この方法では、対話が必要ときにクライアントプログラムをダウンロードする事で、クライアントプログラムの事前配布の問題を解決している。

【0005】

【発明が解決しようとする課題】上述の従来方法1では、ユーザの行う操作に応じた表示画面の遷移はサーバ側で動作するプログラムにより決められる。その為、サーバクライアント間の通信において即時応答性が必要であり、ユーザとの対話中は通信セッションを保つ必要がある。また、従来方法2では、クライアントプログラム事前配布の問題を解決しているが、やはり従来方法1と同様に通信における即時応答性が要求される。

【0006】本発明の目的は、電子メール等即時応答性の低い通信プラットフォームにおいて、サーバで動作するプログラムとユーザとの対話の方法を実現しつつ、同時にクライアントプログラムの事前インストールを不要とする事である。

【0007】

【課題を解決するための手段】この目的を達成するため、請求項1に係る発明は、異なる計算機であるサーバ

(3)

特開平10-149326

計算機とクライアント計算機とをネットワークに接続すると共に、各計算機はネットワークを介してプログラムやデータを相互に転送するための送受信手段を備えたシステムに適用するリモートプログラムユーザ対話方法であって、前記サーバ計算機で実行される主処理プログラム及び前記クライアント計算機で実行されるユーザインタフェース表示プログラムからなる複数プログラムと、それらのプログラム実行時に参照変更される複数データブロックと、管理情報を保持するプロファイルデータ部とから構成される、エージェントと呼ぶソフトウェア単位を定義して予め前記サーバ計算機側に用意しておき、ユーザの操作する前記クライアント計算機において、前記サーバ計算機で実行される主処理プログラムとの対話を行う場合に、前記サーバ計算機から前記クライアント計算機に、エージェント内のユーザインタフェース表示プログラムと、該ユーザインタフェース表示プログラム実行時に参照変更されるデータブロックと、プロファイルデータ部とを備えたサブエージェントを送受信手段により送信し、該サブエージェントを受信した前記クライアント計算機では、該サブエージェントのユーザインタフェース表示プログラムをクライアント内で実行してユーザインタフェース画面を表示することでユーザと直接対話し、ユーザ操作により得られた情報を該サブエージェントのデータブロックへ保存し、その後、該サブエージェントを前記サーバ計算機へ返送し、前記サーバ計算機では、返送された該サブエージェントのデータブロックに格納された変更後のデータを用いて、主処理プログラムを実行することとを特徴とする。

【0008】請求項2に係る発明は、請求項1において、前記エージェントの複数プログラムは、各々が異なる処理系で実行されるが同一の動作を行う異なる形式で記述された複数のユーザインタフェース表示プログラムを含み、前記サーバ計算機から前記クライアント計算機にサブエージェントを送信する際には、前記クライアント計算機で実行可能な形式のユーザインタフェース表示プログラムをサブエージェントに含めて転送するようにしたことを特徴とする。

【0009】請求項3に係る発明は、請求項1または2において、前記エージェントのデータブロックは、当該エージェントの動作属性定義情報を含み、主処理プログラムの動作の前記クライアント計算機での設定変更を行うことによりプログラム動作カスタマイズを行うようにしたことを特徴とする。

【0010】請求項4に係る発明は、請求項1から3のいずれか1つに記載のリモートプログラムユーザ対話方法を実現するプログラムを記憶した記憶媒体である。

【0011】

【発明の実施の形態】以下、図面を用いて、本発明の実施の形態を詳細に説明する。

【0012】図1は、本発明に係るリモートプログラム

ユーザ対話方法を実現するシステムの全体の構成を示したものである。本実施形態のシステムは、サーバ計算機101及びクライアント計算機102が、ネットワーク103により接続されている構成をとる。図2は、図1のシステムのサーバ計算機101及びクライアント計算機102の具体的なハードウェア構成を示す。サーバ計算機101及びクライアント計算機102は、共に図2の構成を有している。

【0013】図1及び図2において、サーバ計算機101では、サーバ側エージェント管理手段111、エージェント実行手段112、及びエージェント送受信手段113がメモリ202に読み込まれ、CPU201により実行される。クライアント計算機102では、クライアント側エージェント管理手段114、エージェント実行手段112、エージェント送受信手段113、表示手段115、及び入力手段116がメモリ202に読み込まれ、CPU201により実行される。表示手段115は表示装置205により表示を行い、入力手段116は入力装置206からの入力を受取る。また、サーバ側エージェント管理手段111とクライアント側エージェント管理手段114との間、及びサーバのエージェント送受信手段113とクライアントのエージェント送受信手段113との間では、それぞれ、通信装置204によりネットワーク103を介して通信を行う。

【0014】サーバ計算機101では、補助記憶装置203内に、テンプレート管理テーブル150及びサーバ側エージェント管理テーブル151が格納される。また、テンプレート格納部152及びエージェント格納部153が、補助記憶装置203内に用意される。テンプレート格納部152及びエージェント格納部153は、それぞれ、テンプレート及びエージェントの格納を行う。

【0015】クライアント計算機102では、補助記憶装置203内に、クライアント側エージェント管理テーブル154が格納される。また、エージェント格納部153が、補助記憶装置203内に用意される。エージェント格納部153は、サーバ101から送られたサブエージェント801の格納を行う。

【0016】ここで本実施形態の概要を説明する。本実施形態では、サーバにおいて処理を行うプログラム（主処理）と、クライアントにおいてユーザインタフェースの表示を行うプログラムと、これらのプログラムの実行時に参照される複数のデータブロックと、管理情報とを纏めたものをエージェントと呼ぶ一つのソフトウェア単位として扱う。各計算機には、計算機内のエージェントを管理するエージェント管理手段と、ネットワークに接続された複数の計算機間でエージェントを転送するエージェント送受信手段と、各計算機内においてエージェントの有するプログラムを実行するエージェント実行手段とを有する。ユーザが操作する計算機（クライアント）と

(4)

特開平10-149326

異なる計算機（サーバ）内に存在するエージェントとユーザとの間で、ユーザインタフェース画面を表示してのユーザ対話を持つ際には、エージェントの有するユーザインタフェース表示プログラムと、ユーザインタフェース表示プログラムから参照更新されるデータブロックと、管理情報とから構成されるエージェント、すなわち元のエージェントの部分から構成されるサブエージェントを生成し、クライアントへ送信する。サブエージェントをクライアントで受信後、クライアントにおいてエージェント内のユーザインタフェース表示プログラムが実行される。ユーザインタフェース表示プログラムは、サーバから共に転送されたデータブロック内のデータを読み出すことで、ユーザへの情報の提示や、ユーザインタフェース表示プログラム自身の振る舞いの変更が可能で、またデータブロックへユーザ対話により得られた情報を格納する事が出来る。ユーザとの対話終了後、サブエージェントは再度サーバへ転送され、エージェント管理手段によりプロフィールデータ部に格納された管理情報を用いてサブエージェントの基となったエージェントを確定し、サブエージェント内のデータブロックが基となったエージェントへ戻される。サーバで実行される主処理プログラムは戻されたデータブロックを参照する事で、クライアントにおけるユーザとの対話で得られた情報を得ることが出来る。以上により、一旦クライアントへ該サブエージェントが転送されてしまえば、ユーザインタフェース表示プログラムとユーザ間の対話はネットワークによる通信を用いず、また、エージェントの転送には通信の即時応答性は要求されない。また、クライアントプログラムの事前インストールも不要である。更に、エージェント内に同一の処理を行う異なる形式で記述されたユーザインタフェース表示プログラムを複数おき、適宜クライアントで実行可能な形式のプログラムをサブエージェント内に持たせることで、異種雑多なクライアント計算機環境に対応することが出来る。以上で、本実施形態の概要説明を終える。

【0017】次に、本実施形態におけるエージェント及びテンプレートの構造について説明する。

【0018】テンプレートはエージェントの生成時にひな形として利用されるものである。本発明はエージェントの生成手順と直接関係するものではないが、本実施形態においては自己完結性のあるシステムを提示するため、生成方法の一例としてテンプレートを利用したものを採用し説明する。

【0019】図3は、エージェント及びそのテンプレートの構成の例を示したものである。本実施形態ではエージェントとテンプレートは同じ構成であり、エージェントはテンプレートを複製することにより生成される。エージェント及びテンプレートは、プロフィールデータ部311、複数のプログラム312～314、及び複数のデータブロック321～323からなる。各プログラム

312～314はエージェント実行手段112により実行され、各データブロック321～323は各プログラム312～314から参照更新されデータの読み書きが行われる。

【0020】複数のプログラム312～314は、サーバで実行される主処理部312、及びクライアントで実行されるユーザインタフェース表示部313～314から構成される。各ユーザインタフェース表示部313～314は、同様の機能を持つ異なる形式のプログラムである。エージェント実行手段112により実行されるプログラムの形式がクライアント毎に異なる場合でも、様々な形式のユーザインタフェース表示部313～314を用意しておくことにより、クライアントに合った形式のプログラムを選択してサーバからクライアントに送信することが出来る。

【0021】図4は、図3のプロファイルデータ部311の構成を示したものである。エージェントID411は、ネットワーク内で一意にエージェントを識別する識別子である。（ネットワーク内で一意の識別子は、計算機のネットワークアドレスと現在時刻の組み合わせ等を利用して生成することが出来る。）エージェント名412は、ユーザが個々のエージェントを識別し易くするためにユーザ自身で付与する名前である。エージェントID411及びエージェント名412は、新規エージェント生成時に付与され、テンプレートにおいては空白である。テンプレートID413は、エージェントの生成の基になったテンプレートを一意に識別する識別子である。テンプレート名414は、エージェントの生成の基になったテンプレートの、ユーザに対して表示される名前である。サーバアドレス415は、エージェントの存在するサーバ101のネットワークアドレスで、クライアント102へ移動したサブエージェントをサーバ101へ返送する際に使われる。プログラム管理テーブル416は、エージェントに含まれる各プログラムに関する情報を格納するテーブルである。

【0022】図4のプロファイルデータ部311において、テンプレートID413、テンプレート名414、及びプログラム管理テーブル416の値は、テンプレートに付与され、エージェント生成時にはテンプレートから複製されることによりエージェントへ引き継がれる。

【0023】図5は、プログラム管理テーブル416の構造を示したものである。プログラム管理テーブル416内には、当該エージェントに含まれる各プログラム毎に、プログラムID501、処理名502、形式503、及び参照データブロックID504、の各フィールドからなるレコードが含まれる。プログラムID501は、当該エージェントに含まれるプログラムを一意に識別する識別子である。処理名502は、当該プログラムが行う処理の種類を示している。当該エージェントがサーバで実行する処理（主処理）を記述したプログラムを

(5)

特開平10-149326

表す "main" と、クライアントで実行する処理 (ユーザインタフェース表示) を記述したプログラムを表す "GUI" との、2種類の値をとる。形式503は、当該プログラムの記述形式を示しており、クライアント102のエージェント実行手段112で実行可能な形式のプログラムを選択するのに利用する。参照データブロックID504には、当該プログラムの実行時に参照されるデータブロックのIDが入っている。プログラムIDとデータブロックIDは共にエージェント内で一意に識別可能ならばよく、図3、図4では単純な通し番号とした。

【0024】図6は、サーバ101内に存在するエージェントのリストを保持するサーバ側エージェント管理テーブル151の構造を示したものである。エージェント毎に、エージェントID601、状態602、アドレス603、及びエージェント名604の各フィールドからなるレコードが含まれる。エージェントID601及びエージェント名604は、各エージェントのプロファイルデータ部311に保持されている値と同じである。状態602は、当該エージェントの実行状態を表すフィールドで、サーバ101のエージェント実行手段112により実行されている状態 "active" と、実行が停止されている状態 "sleep" とがある。アドレス603は、ユーザインタフェースを表示するサブエージェントがクライアント102へ移動している場合の該クライアント102のネットワークアドレスを示す。

【0025】図7は、サーバ101内に登録されているテンプレートのリストを保持するテンプレート管理テーブル150の構造を示したものである。テンプレート管理テーブル150には、テンプレート毎に、テンプレートを一意に識別するテンプレートID701、及びユーザに表示する際に利用されるテンプレート名702の各フィールドからなるレコードが格納される。

【0026】図9は、クライアント102内に存在するサブエージェントのリストを保持するクライアント側エージェント管理テーブル154の構造を示したものである。クライアント102のエージェント管理テーブル154は、ユーザインタフェース表示サブエージェントのアドレスを管理する必要がなく、エージェントID901、状態902、及びエージェント名903の各フィールドから構成される。

【0027】次に、本実施形態において行われる処理手順を、サーバ101に新規にエージェントを生成し、生成したエージェントの動作カスタマイズをクライアント102においてユーザインタフェース画面を表示することで行う応用例に沿って、詳細に説明する。

【0028】新規エージェントの生成手順は、クライアント側エージェント管理手段114からサーバ側エージェント管理手段111に対して、エージェント生成要求メッセージを送信することにより開始される。エージェントの生成はひな形となるテンプレートを指定して行

う。クライアント102側で予めテンプレートのIDを取得しておく必要があるが、これには例えば、クライアント側エージェント管理手段114がサーバ側エージェント管理手段111に対しテンプレートリスト取得メッセージを送信し、サーバ側エージェント管理手段111がテンプレート管理テーブル150 (図7) のデータを返信する手続きによれば良い。後は、クライアント側エージェント管理手段114が取得したテンプレートのリストを表示手段115により表示し、ユーザは入力手段116を用いて、表示されたテンプレートリストから、生成するエージェントのテンプレートを選択すれば良い。これに応じて、クライアント側エージェント管理手段114からサーバ側エージェント管理手段111に対しエージェント生成要求メッセージを送信する。

【0029】エージェント生成要求メッセージには、テンプレートID、エージェント名、及びクライアントのエージェント実行手段112で実行可能なプログラム形式が、パラメータとして併せて付加される。

【0030】どの時点でテンプレートのリストを取得するかについては、利用する通信手段の提供するクライアントサーバ間の応答即時性等により方法を変える必要がある。しかし、テンプレート管理テーブル150の更新頻度はそれ程高くないため、電子メール等の応答即時性が得られない通信手段を利用してテンプレートのリストを取得すればよい。具体的には、例えば、クライアント102の補助記憶装置203内に前回取得したテンプレートリストを保管するようにし、システムの毎起動時にテンプレートリスト取得メッセージをサーバ101に送信し、サーバ101から返送されたテンプレート管理テーブル150のデータを受信して補助記憶装置203内のテンプレートリストを更新する方法等で十分対応できる。

【0031】図10は、エージェント生成要求メッセージをサーバ側エージェント管理手段111が受信した際に、サーバ側エージェント管理手段111が行う処理をフローチャートで記述したものである。まず、エージェント生成要求メッセージを受信する (ステップ1001)。前述の通り、エージェント生成要求メッセージには、テンプレートID、エージェント名、及びクライアント102のエージェント実行手段112で実行可能なプログラム形式が付加されている。次に、受信したエージェント生成要求メッセージで指定されたテンプレートIDで識別されるテンプレートをテンプレート格納部152から取得し、エージェント格納部153へ複製する (ステップ1002)。新規にエージェントIDを生成し (ステップ1003)、生成したIDとサーバ (自機) のネットワークアドレスと受信したエージェント生成要求メッセージにおいて指定されたエージェント名とを、生成するエージェントのプロファイルデータ部311 (詳しくは、エージェントID411、エージェント

(6)

特開平10-149326

名412、及びサーバアドレス415)へ書き込む(ステップ1004)。

【0032】次に、生成するエージェントのレコードをエージェント管理テーブル151へ追加し(ステップ1005)、追加したレコードのエージェントの状態602を停止状態“sleep”に設定する。ここまでで新規エージェントの生成は完了する。その後、生成したエージェントのユーザインタフェース表示サブエージェントを生成し、エージェント送受信手段113によりクライアント102(エージェント生成要求メッセージを送信したクライアント)へ送信する(ステップ1007)。ステップ1007の処理の詳細は、図11のフローチャートを用いて別途後述する。最後に、エージェント管理テーブル151内の送信されたエージェントのアドレス603に当該クライアント102のネットワークアドレスを設定する(ステップ1008)。以上で、サーバ101のエージェント管理手段111がエージェント生成要求メッセージを受信した際に行う処理を完了する。

【0033】図11は、ステップ1007の処理を詳細に説明するフローチャートである。まず、エージェントのプロファイルデータ部311のプログラム管理テーブル416から、処理名“GUI”で、指定された形式(エージェント生成要求メッセージで指定されたプログラム形式)のプログラムを確定する(ステップ1101)。確定したプログラムから参照されるデータブロックを、同じくプログラム管理テーブル416の参照データブロックフィールド504により確定する(ステップ1102)。次に、エージェントのプロファイルデータ部311と確定したプログラムと確定したデータブロックとから構成されるサブエージェントを生成する(ステップ1103)。最後に、エージェント送受信手段113により、生成したサブエージェントを指定アドレス(エージェント生成要求メッセージを送信したクライアントのネットワークアドレス)へ送信する(ステップ1104)。

【0034】図8に、図3で示されるエージェントの構成と、図5で示されるプログラム管理テーブルの状態から、形式“lang-A”で記述された、処理名“GUI”のサブエージェントが生成された場合の構成を示す。プログラム管理テーブル416より、処理名が“GUI”で且つ形式が“lang-A”のプログラムのID02と、参照データブロックのID01が確定する。確定したID02のプログラム313と、ID01のデータブロック321と、プロファイルデータ部311とから生成されたサブエージェントが図8の801である。

【0035】図12は、クライアント102においてサブエージェントを受信した際に、クライアント側エージェント管理手段114が行う処理を示すフローチャートである。エージェント送受信手段113により図8に示すようなサブエージェントを受信し、受信したサブエー

ジェントをエージェント格納部153へ格納する(ステップ1201)。受信サブエージェントのプロファイルデータ部311からエージェントID411とエージェント名412を読み出し(ステップ1202)、読み出したエージェントIDとエージェント名からなるレコードをエージェント管理テーブル154へ追加する(ステップ1203)。追加したエージェント管理テーブル154のレコードのエージェントの状態902を、停止状態“sleep”に設定する(ステップ1204)。

【0036】次に、クライアント102で受信したサブエージェント(ユーザインタフェース表示サブエージェント)801を実行する。クライアント102でのサブエージェントの実行には、サブエージェント受信時に自動的に行なう方法、または、クライアント側エージェント管理手段114により表示されるサブエージェントのリストから実行するサブエージェントをユーザが選択することにより行なう方法がある。このとき重要なことは、ユーザインタフェースの表示に必要なプログラムと、該プログラムの実行時に参照更新されるデータブロックが、クライアント102内に転送され、エージェント格納部153に格納されている事である。このため、サブエージェントのユーザインタフェース表示プログラムの実行時、即ちユーザインタフェース画面表示時には、クライアント102とサーバ101間でネットワークを介した通信を行う必要が全くなく、また、対話の状態をデータブロックへ保存して置くことで、ネットワークと接続せずに対話状態を保持したままユーザインタフェースプログラムの実行を中断し再開する事も可能となる。

【0037】クライアント102でのサブエージェントの実行は、実行するサブエージェントのIDと処理名“GUI”を指定してクライアントのエージェント実行手段112を呼び出すことにより行われる。図13に、クライアント102において、エージェント実行手段112によりサブエージェントを実行する際の処理手順を示す。まず、指定されたエージェントIDのサブエージェントのプログラム管理テーブル416(エージェント格納部153に図8のようなサブエージェントが格納されているので、そのプロファイルデータ部311のプログラム管理テーブル416を参照する)から、指定された処理名“GUI”で且つ形式が当該クライアント102のエージェント実行手段112により実行可能なプログラムを検索する(ステップ1301)。次に、該当するプログラムを読み込み、実行を開始し(ステップ1302)、エージェント管理テーブル154内の当該エージェントの状態902を、活動状態“active”に設定する(ステップ1303)。ここで、プログラムの実行開始時に、実行されるサブエージェント自身のエージェントIDと参照データブロックIDとをプログラムに渡すことにより、プログラム内で読み書き可能なデータブロックを識別出

(7)

特開平10-149326

来る。実行においては、プログラム自身が直接クライアント計算機102のメモリ202に読み込まれCPU201で実行する方法や、エージェント実行手段112がプログラムの解釈実行を行う方法等があるが、何れの方法でも構わない。プログラムの実行開始後、その実行終了を待ち(ステップ1304)、実行終了後にエージェント管理テーブル154内の当該エージェントの状態902を、停止状態"sleep"にする(ステップ1305)。

【0038】サーバ101で新規に生成されたエージェントの場合、クライアント102側では、ユーザインタフェース表示プログラム実行時に、該プログラムからデータブロックの値を読み出し、テンプレートの段階で与えられた初期値であることを判別することで、送られてきたサブエージェントがサーバ101で新規に生成されたものであることを認識できる。サーバ101で新規に生成されたエージェントの場合、クライアント102のユーザインタフェース表示プログラムは、最初に動作カスタマイズの画面を表示する(そのようにプログラムを作成しておく)。この場合、当該ユーザインタフェース表示プログラムが使用するデータブロックは、当該エージェントの動作属性定義情報を含むものであり、その設定を行うことにより、主処理プログラムの動作のクライアントでの設定変更(すなわち、プログラム動作カスタマイズ)を行うことができるものである。ユーザは、表示されたユーザインタフェースの画面を介してエージェントの動作設定(カスタマイズ)を行い、当該サブエージェントをサーバ101へ返送する。

【0039】サーバ101へのサブエージェント801送信のユーザ指示は、送信するサブエージェントをクライアント102のエージェント管理手段114が表示する画面からユーザが選択することにより行なう方法、またはサブエージェントのユーザインタフェース表示プログラムの表示する画面から指示する方法等がある。サブエージェントのユーザインタフェース表示プログラムの表示する画面から指示する場合、ユーザインタフェース表示プログラム内に、クライアント側エージェント管理手段114に対し送信を指示する手順を予め組み込んでおく必要がある。

【0040】図14は、クライアント側エージェント管理手段114がサーバ101へサブエージェント801を送信する(送り返す)際の処理を示すフローチャートである。まず、送信するサブエージェント801のプロファイルデータ部311から送信元サーバのネットワークアドレス415を取得する(ステップ1401)。次に、エージェント送受信手段113により、取得したサーバ101のアドレスへサブエージェント801を送信する(ステップ1402)。最後に、エージェント格納部153から、送信したサブエージェント801を削除し(ステップ1403)、エージェント管理テーブル1

54から、送信したサブエージェントのレコードを削除する(ステップ1404)。

【0041】次に、図15のフローチャートを用いて、上記のように返送されたサブエージェント801をサーバ101で受信した際に、サーバ側エージェント管理手段111が行う処理ステップを説明する。始めに、エージェント送受信手段113によりサブエージェント801を受信し(ステップ1501)、受信したサブエージェント801のプロファイルデータ部311よりエージェントID411を取得する(ステップ1502)。次に、取得したエージェントIDのレコードをエージェント管理テーブル151から検索する(ステップ1503)。検索したレコードのエージェントの実行状態602をチェックし(ステップ1504)、エージェントが実行状態"active"の場合、エージェント実行手段112により当該エージェントの主処理プログラム312の実行を停止する(ステップ1508)。この際、エージェント実行手段112により、エージェント管理テーブル151のエージェントの状態602は停止状態"sleep"に設定される。本例の方法では、サーバ101は、エージェントを"active"にしたままサブエージェントをクライアント102に送ることができるため、サブエージェントがクライアント102から返送されてきたときは、一旦当該エージェントを"sleep"にした後、次のステップ1505でデータブロックの書き替えを行うようにしている。

【0042】次に、受信したサブエージェント801のデータブロックを、エージェント格納部153に格納された同一IDのエージェントの同一IDのデータブロックに上書きする(ステップ1505)。次に、ステップ1503において検索したエージェント管理テーブル151のレコードのアドレス603を空白にする(1506)。最後に、受信したサブエージェントのID、処理名"main"、当該サーバ101で実行可能なプログラムの形式を指定し、エージェント実行手段112により、受信したエージェントの主処理プログラム312の実行を開始する(ステップ1507)。なお、サーバ101のエージェント実行手段112によるエージェントの実行ステップは、図13を用いて既に説明したクライアント102におけるサブエージェントの実行ステップと同様である。ただし、クライアント102における実行ではエージェント管理テーブル154を用いたが、サーバ101における実行ではエージェント管理テーブル151を用いるものとする(ステップ1303、1305)。

【0043】エージェントの主処理プログラム312は、クライアント102で設定されたデータブロック321の内容を参照し、ユーザの指定に従い動作を行うことが出来る。以上で、サーバ101に新規生成したエージェントの主処理プログラム動作の、クライアント10

(8)

特開平10-149326

2からのカスタマイズを完了する。サーバ101のエージェント管理テーブル151には生成されたエージェントが登録され(状態602は、ユーザの指示に応じて"sleep"の場合もあるし"active"の場合もある)、エージェント格納部153には生成されたエージェント(クライアント102からのカスタマイズが完了しているもの)が格納されたことになる。

【0044】次に、サーバ101で既に生成されたエージェントの実行中の主処理プログラム312とのユーザ対話(クライアント102からの対話)の手順について説明する。

【0045】ユーザ対話は、クライアント102のエージェント管理手段114からサーバ101のエージェント管理手段111へ、エージェント転送要求メッセージを送信することにより開始される。エージェント転送要求メッセージには、クライアント102への転送の対象となるエージェントのID、クライアント102のエージェント実行手段112で実行可能なプログラムの形式、及びエージェント転送後のサーバ101でのエージェントの動作を指定する転送後動作指定フラグがパラメータとして付加される。転送後動作指定フラグは、クライアント102へユーザインタフェース表示サブエージェントを転送した後もサーバ101側のエージェントの主処理プログラム312の実行を継続するか否かを指定するフラグで、値として動作継続"continue"と停止"stop"の2種類を設ける。

【0046】クライアント側エージェント管理手段114から上記エージェント転送要求メッセージの送信を行う際には、事前にサーバ101に存在するエージェントのIDを取得しておく必要があるが、テンプレートの場合と同様にして、サーバ側エージェント管理手段111とクライアント側エージェント管理手段114間の通信処理により、サーバ101のエージェント管理テーブル151の情報を取得する方法、またはクライアント側の補助記憶装置203内にサーバ101に存在するエージェントの情報を保持するテーブルを用意しサーバ101上のエージェントに対するクライアント102からの操作の履歴を用いて更新する方法等が利用できる。電子メール等の即時応答性の得られない通信手段を利用する場合は後者を用いる事が出来る。

【0047】図16は、サーバ101のエージェント管理手段111がエージェント転送要求メッセージを受信した際の処理手順を示すフローチャートである。前述したとおり、エージェント転送要求メッセージの受信において、エージェントID、クライアント102のエージェント実行手段112で実行可能なプログラムの形式、及びサブエージェント転送後のサーバ101でのエージェントの主処理プログラム実行を指定する転送後動作指定フラグの各情報がパラメータとして渡される(ステップ1601)。エージェント管理テーブル151から、

指定されたIDのエージェントを検索し(ステップ1602)、エージェントのアドレス603が空白かどうかをチェックする(ステップ1603)。アドレス603が設定されている場合は、既に何れかのクライアントへサブエージェントを送信しているということであるから、要求元のクライアント102のエージェント管理手段114へエラーメッセージを送信して(ステップ1608)、終了する。

【0048】ステップ1603でアドレス603が空白であったら、エージェント管理テーブル151のエージェントの状態602と転送後動作指定フラグの値をチェックし(ステップ1604)、転送後動作指定フラグが停止"stop"で且つエージェントの状態602が"active"の場合、ステップ1607を実行する。ステップ1607では、エージェント実行手段112により当該エージェントのプログラム(主処理)の実行を停止する。このとき、エージェント管理テーブル151のエージェントの状態602は、停止"sleep"へ設定される。次に、エージェントのユーザインタフェース表示サブエージェントを生成し、要求元のクライアント102へ送信する(ステップ1605)。ステップ1605の処理の詳細は図10のステップ1007と同様で、図11を用いて前述した通りである。最後に、エージェント管理テーブル151の送信エージェントのアドレス603に、要求元のクライアント102のネットワークアドレスを設定する。

【0049】以降、サブエージェントを受信したクライアント102の処理手順、サブエージェントのサーバ101への返送手順、サブエージェントを受信したサーバ101での処理手順は、エージェントの新規生成での例と同様である。

【0050】

【発明の効果】以上に述べたように、本発明によれば、他計算機で動作するプログラムとのユーザ対話を行う際に、クライアントへユーザインタフェース表示プログラムと該プログラムから参照更新されるデータブロックとを併せて送信し、クライアント内でのユーザインタフェース表示プログラムの実行時にはネットワークを介した通信を行わない。これにより、電子メール等の即時応答性の低い通信プラットフォームで接続されたネットワーク環境や、携帯端末の様に常時ネットワーク接続が困難な計算機環境において、ユーザの操作する計算機と異なる計算機上で動作するプログラムと対話を行う機能を提供することができる。また、実行状態をデータブロックへ保存することで、ネットワークへ再接続する事なく、対話プログラム実行の中断及び再開が可能となる。また、対話実行時にクライアントへユーザインタフェース表示プログラムと参照データブロックとを送信する為、クライアントプログラムの事前インストールを不要とする。更に、複数の形式で記述された同一動作を行うユー

(9)

特開平10-149326

ザインタフェース表示プログラムをエージェントに持たせ、クライアントへ送信する際にクライアントの計算機環境に合ったプログラムを適宜選択することで、異種雑多なクライアント計算機環境に対応することが出来る。

【図面の簡単な説明】

【図1】本発明に係るリモートプログラムユーザ対話方法を実現するシステムの機能ブロック図

【図2】図1のシステムのサーバ計算機及びクライアント計算機の具体的なハードウェア構成図

【図3】エージェント、及びテンプレートの構成図

【図4】プロファイルデータ部の構成図

【図5】プログラム管理テーブルの構造説明図

【図6】サーバ側エージェント管理テーブルの構造説明図

【図7】クライアント側テンプレート管理テーブルの構造説明図

【図8】ユーザインタフェース表示サブエージェントの生成例における構成説明図

【図9】クライアント側エージェント管理テーブルの構造説明図

【図10】エージェント生成要求メッセージ受信時のエージェント管理手段処理フロー図

【図11】ステップ1007の処理を詳細に説明する処理フロー図

【図12】クライアント側エージェント受信時のエージェント管理手段処理フロー図

【図13】エージェント実行時のエージェント実行手段処理フロー図

【図14】クライアント側エージェント送受信手段のエージェント送信処理フロー図

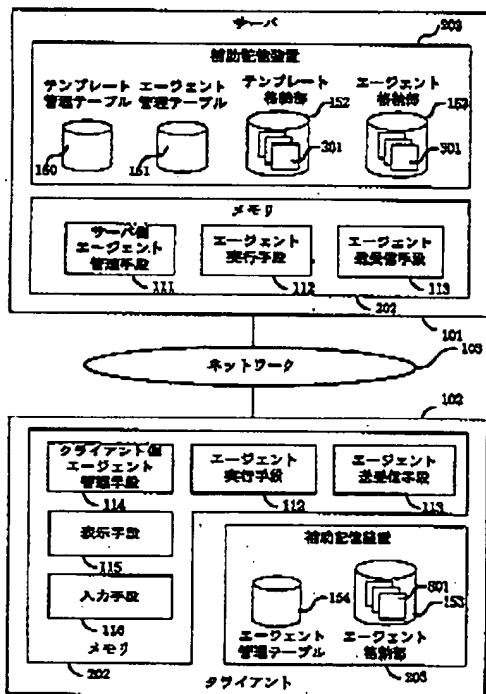
【図15】クライアント側エージェント受信時のエージェント管理手段処理フロー図

【図16】サーバ側エージェント受信時のエージェント管理手段処理フロー図

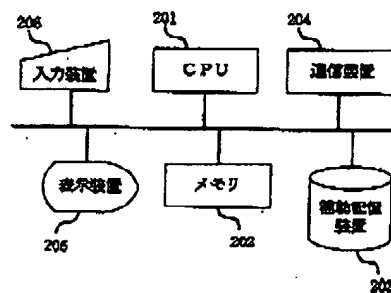
【符号の説明】

101…サーバ、102…クライアント、103…ネットワーク、111…サーバ側エージェント管理手段、112…エージェント実行手段、113…エージェント送受信手段、114…クライアント側エージェント管理手段、115…表示手段、116…入力手段、150…テンプレート管理テーブル、151…エージェント管理テーブル、152…テンプレート格納部、153…エージェント格納部、154…エージェント管理テーブル、301…エージェント、311…プロファイルデータ部、312…主処理プログラム、313～314…ユーザインタフェース表示プログラム、321～323…データブロック。

【図1】



【図2】



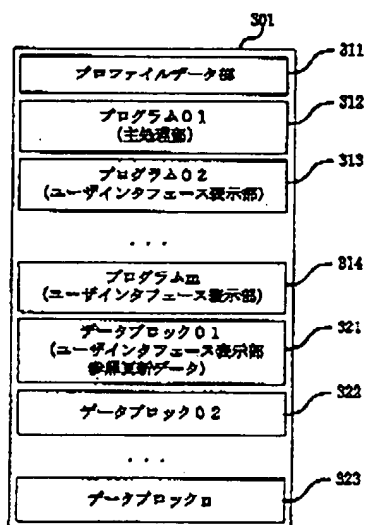
【図7】

テンプレートID	テンプレート名
0001	名前1
0002	名前2
...	...

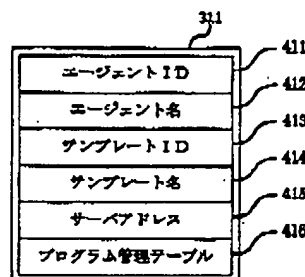
(10)

特開平10-149326

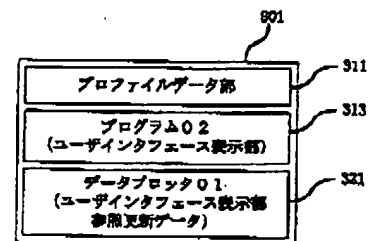
【図3】



【図4】



【図8】



【図6】

エージェントID	状態	アドレス	エージェント名
0001	active	-	名前1
0002	sleep	-	名前2
0003	sleep	client-1	名前3

【図5】

プログラムID	処理名	形式	参照データブロック
01	main	lang-A	01, 02
02	GUI	lang-A	01
03	GUI	lang-B	01

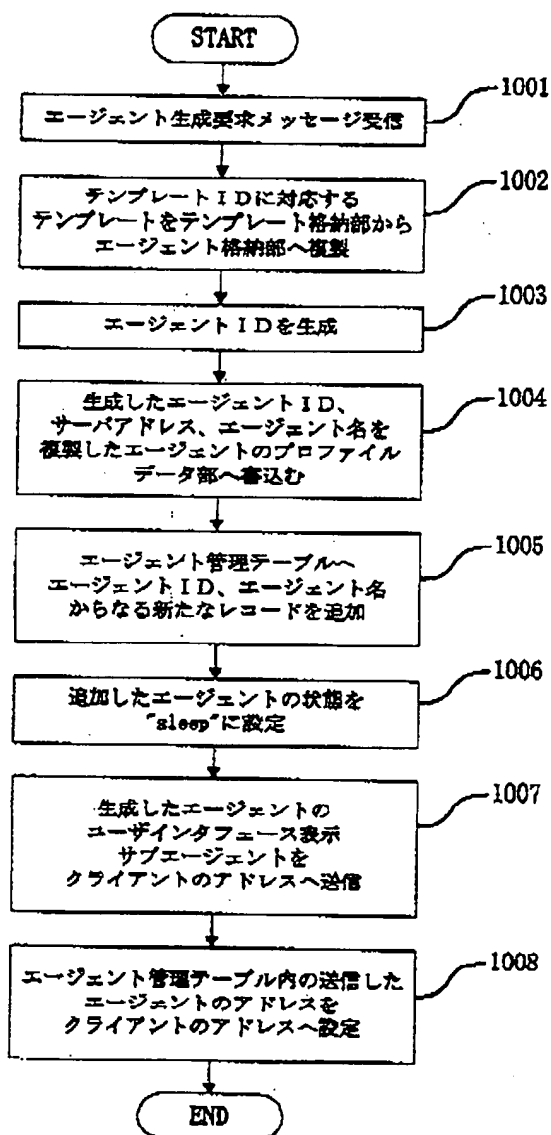
【図9】

エージェントID	状態	エージェント名
0003	active	名前3

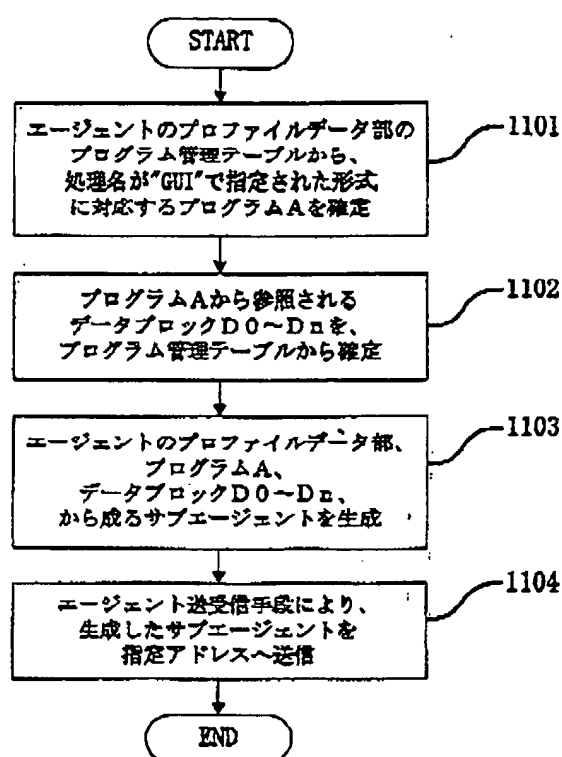
(11)

特開平10-149326

【図10】



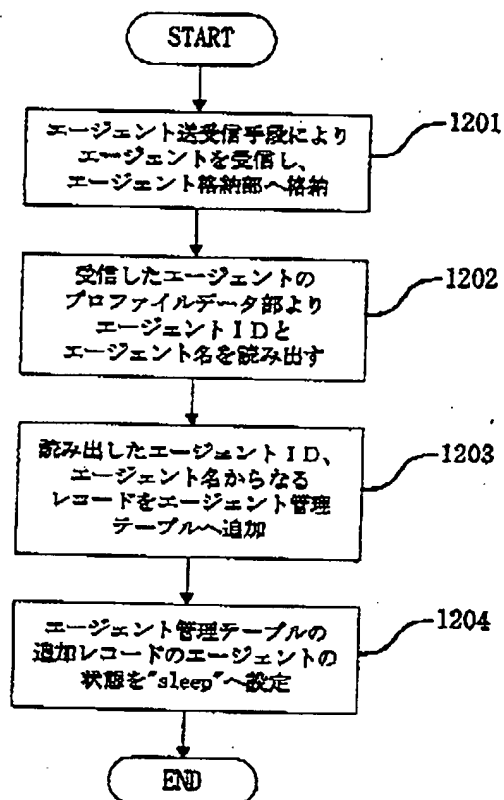
【図11】



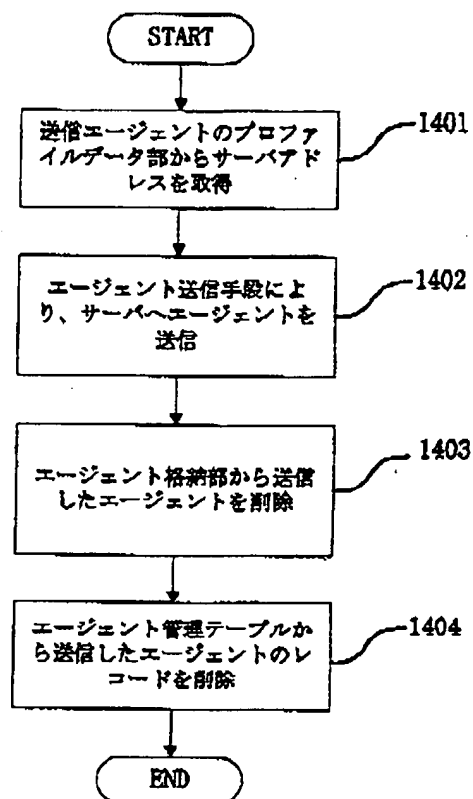
(12)

特開平10-149326

【図12】



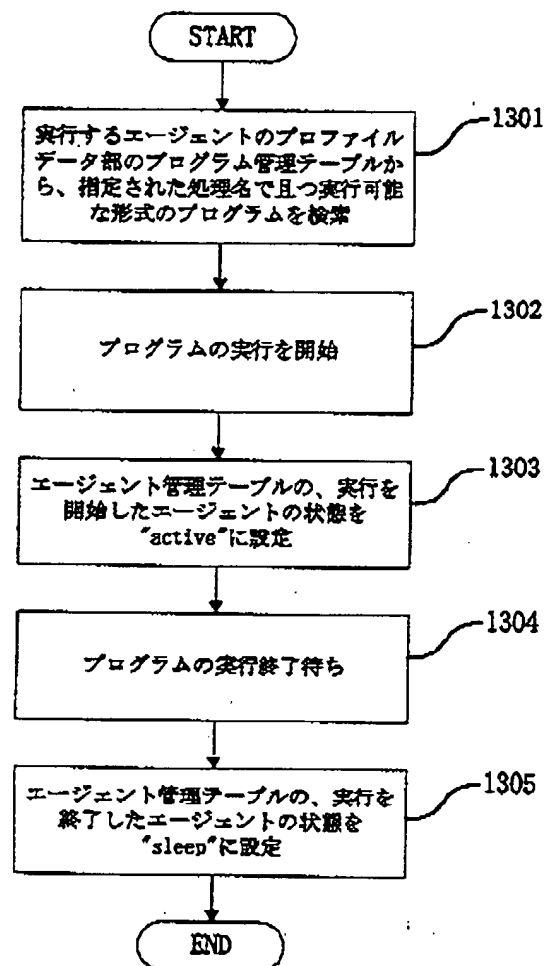
【図14】



(13)

特開平10-149326

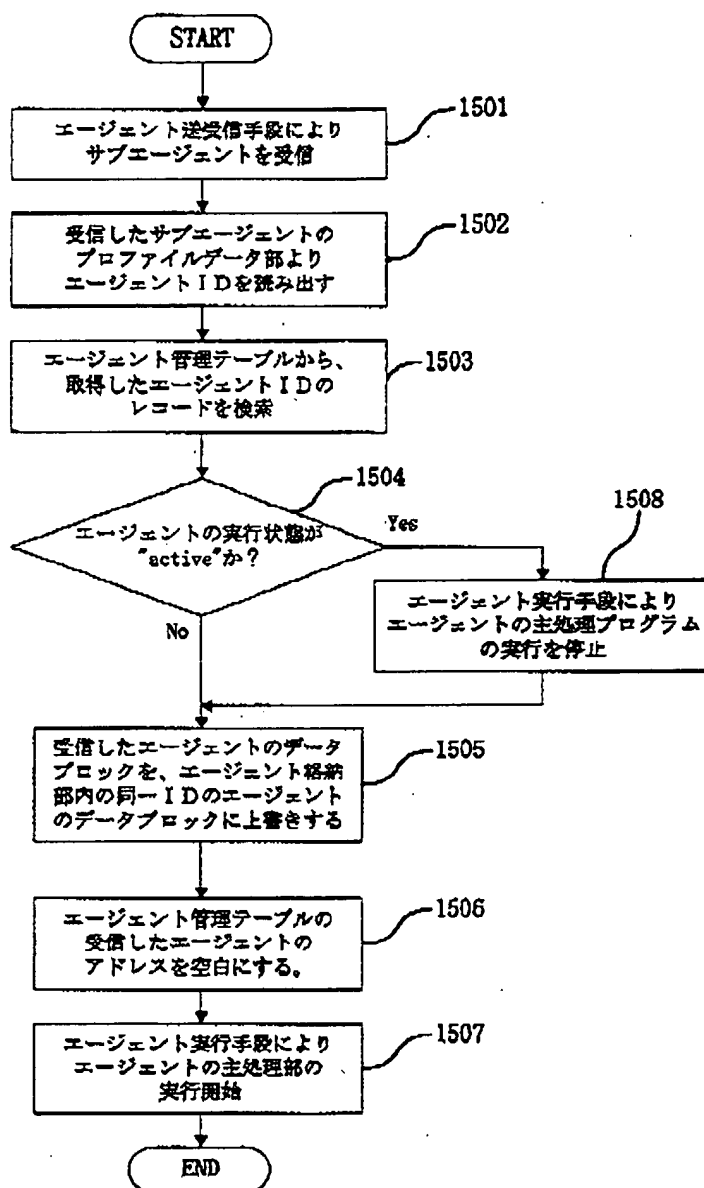
【図13】



(14)

特開平10-149326

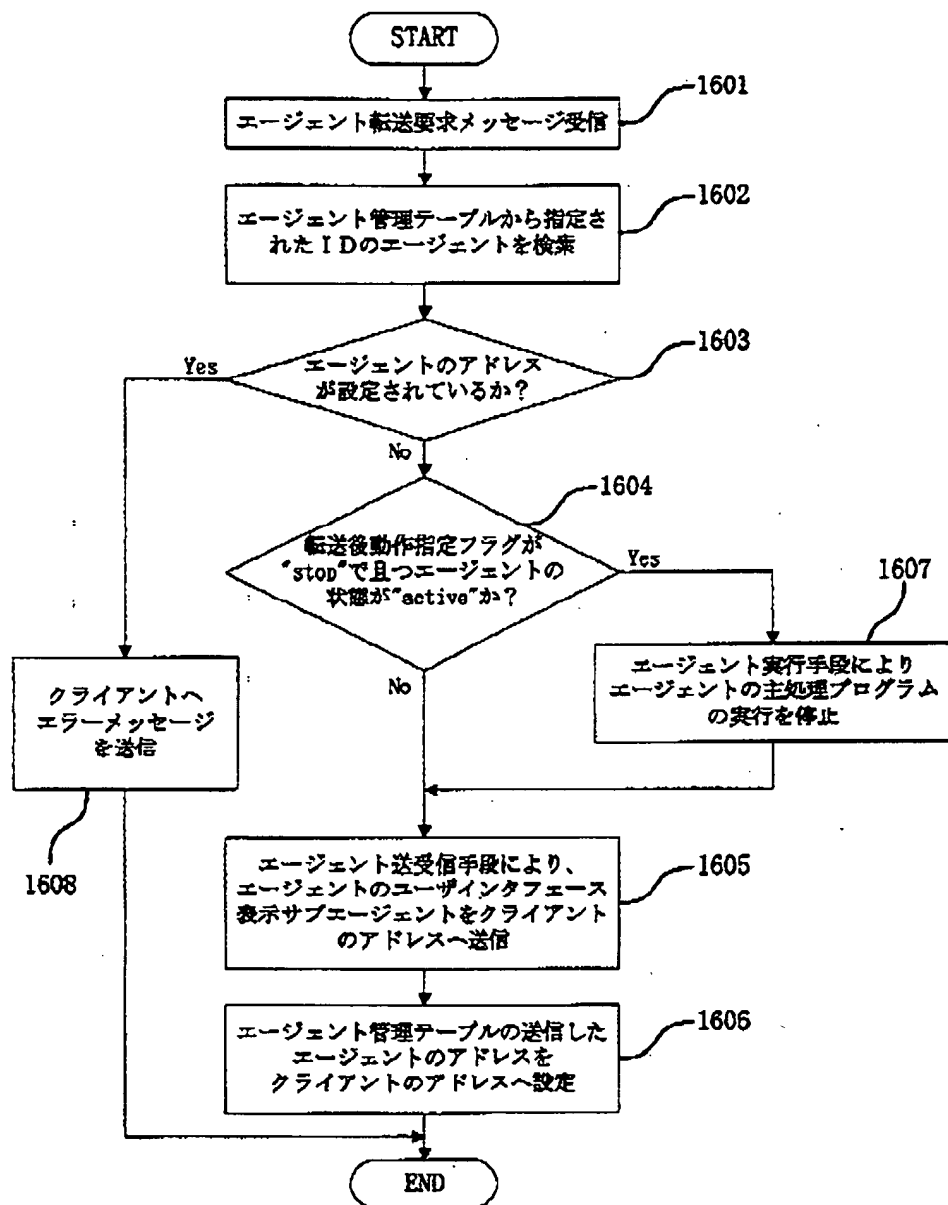
【図15】



(15)

特開平10-149326

【図16】



フロントページの続き

(72)発明者 青島 達人

神奈川県川崎市幸区鹿島田890番地 株式  
会社日立製作所情報・通信開発本部内